

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 11-053049

(43)Date of publication of application : 26.02.1999

(51)Int.Cl. G06F 1/04

G06F 1/32

G06F 1/26

(21)Application number : 09-210627 (71)Applicant : TOSHIBA CORP

(22)Date of filing : 05.08.1997 (72)Inventor : MAKI YASUNORI

(54) COMPUTER SYSTEM

(57)Abstract:

PROBLEM TO BE SOLVED: To reduce the power consumption of a computer by enabling control over the stop and restart of a bus clock without mounting any special function on a peripheral device on a bus.

SOLUTION: When a bus monitor circuit 161 detects FRAME# and IRDY# being both supported and a bus request detecting circuit 162 and a system event detecting circuit 163 detect a bus request signal and a system event signal not being generated, the output of a 3-input OR gate 164 goes down to 'L' indicating that the bus is not in operation. Consequently, a clock control signal output circuit 165 generates a clock control signal indicating the stop of PCICLK, and consequently the supply to PCICLK to respective PCI devices is stopped.

LEGAL STATUS [Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

*** NOTICES ***

JPO and INPIT are not responsible for any damages caused by the use of this translation.

1.This document has been translated by computer. So the translation may not reflect

the original precisely.

2.**** shows the word which can not be translated.

3.In the drawings, any words are not translated.

CLAIMS

[Claim(s)]

[Claim 1] A bus clock generation means to generate the bus clock for giving the timing of the transaction on said bus to two or more peripheral devices combined with the bus of a computer system, and the peripheral device of these plurality, A bus idle detection means to supervise the transaction on said bus and to detect whether it is a bus idle condition, An event detection means to detect the existence of the bus demand signal from said each peripheral device, and an interrupt signal, It is a clock control means to control clock generation actuation of said bus clock generation means based on the detection result of said bus idle detection means and said event detection means. The computer system characterized by providing a clock control means to stop said bus clock when it is detected that it is in a bus idle condition and it is detected that said bus demand signal and interrupt signal are not generated.

[Claim 2] After it is detected that it is in said bus idle condition and it is detected that said bus demand signal and interrupt signal are not generated, said clock control means so that said bus clock may be stopped after fixed time amount progress When generating of a means to delay the timing which stops said bus clock, discharge of said bus idle condition, said bus demand signal, or said

interrupt signal is detected, The computer system according to claim 1 characterized by providing a means to make supply of said bus clock resume.

[Claim 3] A bus clock generation means to generate the bus clock for giving the timing of the transaction on said bus to two or more devices combined with the bus of a computer system, and the device of these plurality, A bus idle detection means to supervise the transaction on said bus and to detect whether it is a bus idle condition, An event detection means to detect the existence of the bus demand signal from said each device, and an interrupt signal, It is a clock control means to control clock generation actuation of said bus clock generation means based on the detection result of said bus idle detection means and said event detection means. The computer system characterized by providing a clock control means to reduce the frequency of said bus clock when it is detected that it is in a bus idle condition and it is detected that said bus demand signal and interrupt signal are not generated.

[Claim 4] After it is detected that it is in said bus idle condition and it is detected that said bus demand signal and interrupt signal are not generated, said clock control means so that the frequency of said bus clock may fall after fixed time amount progress When generating of a means to delay the timing to which the frequency of said bus clock is reduced, discharge of said bus idle condition, said bus demand signal, or said interrupt signal is detected, The computer system

according to claim 3 characterized by providing a means to return the frequency of said bus clock to the original condition.

DETAILED DESCRIPTION

[Detailed Description of the Invention]

[0001]

[Field of the Invention] Especially this invention relates to the computer system which has a bus clock control function about a computer system.

[0002]

[Description of the Prior Art] Conventionally, as a system bus used for a personal computer, although the ISA (Industry Standard Architecture) bus and the EISA (Extended ISA) bus were in use, recently, a PCI (Peripheral Component Interconnect) bus is beginning to be adopted for improvement in the speed of a data transfer rate, and construction of the system architecture independent of a processor.

[0003] In the PCI bus, almost all data transfer is based on block transfer, and each [these] block transfer is realized using burst transmission. Thereby, in a PCI bus, the data transfer rate of a maximum of 133 M bytes/second can be realized (when a data bus is 32-bit width of face).

[0004] Therefore, if a PCI bus is adopted, it becomes possible to perform data transfer between I/O devices and between a system memory and an I/O device etc. at a high speed, and system performance can be raised.

[0005] In such a PCI bus architecture, the comparatively high-speed bus clock of 33MHz is used for implementation of high-speed operation. This bus clock is for providing each peripheral device on a bus with the timing of the transaction on a PCI bus, and the peripheral device on a bus is always continuing being supplied during system behavior.

[0006]

[Problem(s) to be Solved by the Invention] However, such a high-speed bus clock is set to one of the big factors which raise the power consumption of a computer. For this reason, recently, the specification ("CLKRUN" protocol) for stopping a bus clock is increasingly incorporated also in the specification of a PCI bus.

[0007] However, in order to mount the this "CLKRUN" protocol in a system, to have a function for all the devices on a PCI bus to correspond to the "CLKRUN"

protocol is needed. For this reason, when the existing peripheral device which does not support the "CLKRUN" protocol existed, the "CLKRUN" protocol could not be used, and a bus clock was not able to be stopped.

[0008] This invention is made in view of such a point, and it enables it to control a bus clock, without mounting special functions, such as the "CLKRUN" protocol, in a peripheral device, and aims at offering the computer system which can aim at reduction of power consumption easily.

[0009]

[Means for Solving the Problem] Two or more peripheral devices by which the computer system by this invention is combined with the bus of a computer system, A bus clock generation means to generate the bus clock for giving the timing of the transaction on said bus to the peripheral device of these plurality, A bus idle detection means to supervise the transaction on said bus and to detect whether it is a bus idle condition, An event detection means to detect the existence of the bus demand signal from said each peripheral device, and an interrupt signal, It is a clock control means to control clock generation actuation of said bus clock generation means based on the detection result of said bus idle detection means and said event detection means. When it is detected that it is in a bus idle condition and it is detected that said bus demand signal and interrupt signal are not generated, it is characterized by providing a clock control means

to stop said bus clock.

[0010] In this computer system, a system state is investigated by detecting the existence of a bus idle condition, a bus demand signal, and an interrupt signal, it is judged that it is in a bus idle condition, and he is a system idle when it is detected that the bus demand signal and the interrupt signal are not generated, and a bus clock is stopped. In this case, since it will not be stopped by the bus clock in the limitation in which the bus demand signal or the interrupt signal is generated even if it is in a bus idle condition, actuation of a peripheral device is not affected. Therefore, an unnecessary bus clock can be stopped without mounting special functions, such as the "CLKRUN" protocol, in a peripheral device, and reduction of power consumption can be aimed at.

[0011] Moreover, after it is detected as said clock control means that it is in said bus idle condition and it is detected that said bus demand signal and interrupt signal are not generated, it is desirable to include a means to delay the timing which stops said bus clock so that said bus clock may be stopped after fixed time amount progress.

[0012] Since time amount will be required by resumption of bus clock supply even if an interrupt signal etc. is generated after that if a bus clock is immediately stopped at the time of a system idle, the fall of system performance will be caused. Therefore, in order to prevent the fall of system performance, it is

required to make it make it perform by summarizing as many processings as possible at the clock days of supply, and this can be realized by delaying the timing which stops a bus clock.

[0013] Moreover, two or more devices with which the computer system of this invention is combined with the bus of a computer system, A bus clock generation means to generate the bus clock for giving the timing of the transaction on said bus to the device of these plurality, A bus idle detection means to supervise the transaction on said bus and to detect whether it is a bus idle condition, An event detection means to detect the existence of the bus demand signal from said each device, and an interrupt signal, It is a clock control means to control clock generation actuation of said bus clock generation means based on the detection result of said bus idle detection means and said event detection means. When it is detected that it is in a bus idle condition and it is detected that said bus demand signal and interrupt signal are not generated, it is characterized by providing a clock control means to reduce the frequency of said bus clock.

[0014] According to this configuration, the frequency of that bus clock falls instead of stopping a bus clock completely. Usually, since the peripheral device is constituted so that it can operate about a part of logic of the interior also in the condition that the bus clock is stopped, and an interrupt signal and a bus demand signal can be generated normally, it is satisfactory, but when a bus

clock is stopped depending on a peripheral device, there are some which cannot operate at all. Therefore, when such a device is connected, it is required to supply a necessary minimum clock rather than to stop a bus clock completely, and to guarantee actuation of these devices. Thereby, it becomes possible to aim at reduction of power consumption, guaranteeing actuation of a device.

[0015]

[Embodiment of the Invention] Hereafter, the operation gestalt of this invention is explained with reference to a drawing. The configuration of the computer system concerning 1 operation gestalt of this invention is shown in drawing 1 . This computer system is a portable computer the notebook type in which a dc-battery drive is possible, or laptop type, and the processor bus 1, PCI bus 2, ISA Bus 3, CPU11, host-PCI bridge equipment 12, main memory 13, the various PCI peripheral devices 14 and 15, the bus clock control circuit 16, the bus clock generation circuit 17, PCI-ISA bridge equipment 18, the various ISA peripheral devices 19 and 20, etc. are formed on that system board.

[0016] The PCI bus clock (PCICLK) generated by the bus clock generation circuit 17 is supplied to the thing 12 which functions as a PCI device connected to PCI bus 2 among these components, i.e., host-PCI bridge equipment, the various PCI peripheral devices 14 and 15, and PCI-ISA bridge equipment 18. This PCI bus clock (PCICLK) gives the timing of a bus transaction to each PCI

device, and the cycle on PCI bus 2 is altogether performed synchronizing with a PCI bus clock (PCICLK).

[0017] Next, the function and configuration of each component of drawing 1 are explained. CPU11 is realized by the microprocessor "Pentium" currently manufactured and sold by U.S. Intel. The processor bus 1 directly linked with the I/O pin of this CPU11 has the data bus of 64-bit width of face. Main memory 13 is a memory device which stores the application program an operating system, a device driver, and for activation, processed data, etc., and is constituted by two or more DRAM modules.

[0018] Host-PCI bridge equipment 12 is the bridge LSI which connects between the processor bus 1 and PCI buses 2, and functions as one of the bus masters of PCI bus 2. This host-PCI bridge equipment 12 has the function to change a bus cycle including data and the address in both directions, the function which carries out the access control of the main memory 13 through a memory bus between the processor bus 1 and PCI bus 2.

[0019] PCI bus 2 is the input/output bus of a clock synchronous type, and as mentioned above, all the cycles on PCI bus 2 are performed synchronizing with a PCI bus clock. The frequency of a PCI bus clock is a maximum of 33MHz. PCI bus 2 has the address/data bus used in time sharing. This address/data bus are 32-bit width of face.

[0020] The data transfer cycle on PCI bus 2 consists of an address phase and one or more data phases which follow it. The address and a transfer type are outputted in an address phase, and data (8 bits, 16 bits, 24 bits, or 32 bits) are outputted in a data phase.

[0021] The PCI devices 14 and 15 are for example, a graphics controller, a PC card (CardBus) controller, an IrDA controller, a SCSI controller, etc., and function as a bus master of PCI bus 2 like host-PCI bridge equipment 12.

[0022] PCI-ISA bridge equipment 16 is the bridge LSI which connects between PCI bus 2 and ISA Buses 3, and functions as one of the PCI devices. The ISA devices 19 and 20 on ISA Bus 3 are HDD, a system timer, a keyboard controller, etc.

[0023] The bus clock control circuit 16 is for controlling a halt/restart of supply of the PCI bus clock (PCICLK) to each PCI device, and control of a halt/restart of the PCI bus clock (PCICLK) is performed based on a bus idle signal, a bus demand signal, and a system-event signal. a bus idle signal -- the operating state of PCI bus 2, i.e., a bus, -- it is shown whether it is working (under bus cycle activation). A bus demand signal is a signal for a bus master to require the royalty of PCI bus 2 of a PCI bus arbiter. A system-event signal shows that a certain event occurred within systems, such as an interrupt signal.

[0024] If the bus clock control circuit 16 detects the idle of the PCI bus by the bus

idle signal and there are not a bus demand signal and a system-event signal, it will control the bus clock generation circuit 17 by the clock control signal, and will stop a bus clock. Moreover, when a bus demand signal or a system-event signal occurs during a bus clock halt, the bus clock control circuit 16 controls the bus clock generation circuit 17 by the clock control signal, and makes supply of a bus clock resume.

[0025] The concrete example of a configuration of the bus clock control circuit 16 is shown in drawing 2 . Like illustration, the bus clock control circuit 16 consists of the bus supervisory circuit 161, the bus demand detector 162, a system-event detector 163, the OR gate 164, and a clock control signal output circuit 165.

[0026] the time of detecting that the bus supervisory circuit 161 supervises the transaction on PCI bus 2 using frame signal FRAME# and initiator ready signal IRDY# for generating the above-mentioned bus idle signal which it is and were defined on PCI bus 2, and does not bus operate -- a bus idle signal -- "H" level -- asserting -- a bus -- DEASATO [a bus idle signal / "L" level] when a working thing is detected.

[0027] Here, FRAME# is a signal driven by the current master, in order to show initiation and the period of a transaction. When DEASATO [FRAME#], it is shown that a transaction is the last data phase. IRDY# is a signal driven by the current master, and in a light cycle, it is asserted in order to show that the master

outputted definite data on the bus, and in a read cycle, it is asserted in order to show what the preparation with which a master receives data was able to carry out.

[0028] When DEASATO [both these FRAME# and IRDY#], the bus supervisory circuit 161 detects that it is in a bus idle condition, and asserts a bus idle signal on "H" level.

[0029] The bus demand detector 162 supervises all BASURI QUEST signal REQ1# inputted into a PCI bus arbiter from each PCI device - REQn#, and when either of REQ1# - REQn# is asserted, it asserts a bus demand signal on "H" level.

[0030] The monitor about mask impossible interrupt signal NMI and system management interrupt signal SMI is performed further, and a system-event detector asserts a system-event signal on "H" level, when one of interrupt signals is generated, interrupt signal INTA-D which interrupts from each PCI device and is inputted into a controller, interrupt signal IRQ 0-15 which interrupts from each ISA device and is inputted into a controller, and.

[0031] The reversal input of the bus idle signal from the bus supervisory circuit 161 is carried out at the 1st input of 3 input OR gate 164. Moreover, the bus demand signal and system-event signal from the bus demand detector 162 and the system-event detector 163 are inputted into the 2nd input and the 3rd input

of 3 input OR gate 164 as it is, respectively. the output of 3 input OR gate 164 -- bus working ("H" level) / bus -- un--- being working ("L" level) -- it is shown and is inputted in the clock control signal output circuit 165.

[0032] The clock control signal output circuit 165 generates the clock control signal which directs a halt/restart of PCICLK based on the output of 3 input OR gate 164. Next, actuation of this operation gestalt is explained with reference to drawing 3 and drawing 4 .

[0033] The timing chart of drawing 3 is the timing in the case of stopping a bus clock (PCICLK). That is, if DEASATO [both FRAME# and IRDY#] is detected and it is detected that the bus demand signal and the system-event signal are not generated, the output of 3 input OR gate 164 will serve as "L" level which shows under bus un-operating. Thereby, from the clock control signal output circuit 165, the clock control signal which directs a halt of PCICLK is generated, and supply of PCICLK to each PCI device is suspended by this.

[0034] If a clock demand signal is generated during the supply interruption period of this PCICLK (detection of assertion of FRAME# by the bus supervisory circuit 161, or IRDY#, detection of generating of the bus demand signal by the bus demand detector 162, or detection of generating of the interrupt signal by the system-event detector 163), the output of 3 input OR gate 164 will serve as "H" level which shows under bus actuation. Thereby, from the clock control

signal output circuit 165, the clock control signal which directs the restart of PCICLK is generated, and supply of PCICLK to each PCI device is resumed by this.

[0035] Thus, according to the configuration of drawing 2 , a system state is investigated by detecting the existence of a bus idle condition, a bus demand signal, and an interrupt signal, it is judged that it is in a bus idle condition, and he is a system idle when it is detected that the bus demand signal and the interrupt signal are not generated, and PCICLK is stopped. In this case, since it will not be stopped by PCICLK in the limitation in which the bus demand signal or the interrupt signal is generated even if it is in a bus idle condition, actuation of each PCI device is not affected. Moreover, if a bus demand signal and an interrupt signal are generated from a PCI device during a halt of PCICLK, supply of PCICLK will be resumed and it will become possible to perform the normal bus transaction which synchronized with PCICLK. Therefore, an unnecessary bus clock can be stopped without mounting special functions, such as the "CLKRUN" protocol, in a PCI device, and reduction of power consumption can be aimed at.

[0036] The 2nd example of a configuration of the bus clock control circuit 16 is shown in drawing 5 . Here, in addition to the configuration of drawing 2 , the snap timer 166 is formed. This snap timer 166 is for delaying the halt timing of PCICLK during a fixed period, and after the output of 3 input OR gate 164 serves as "L"

level which shows under bus un-operating, when count actuation is started and predetermined counted value is reached, it notifies it to the clock control signal output circuit 165. By this, the clock control signal which directs a halt of PCICLK will be generated [time amount / of the snap timer 166 / count]. Counted value of the snap timer 166 can be made programmable by preparing the configuration register for setting up a counter value for example, in the bus clock control circuit 16, and setting up by software the counter value equivalent to the time delay for which it wishes there.

[0037] The situation of the state transition of the bus clock control circuit 16 of drawing 5 is shown in drawing 6 . In drawing 6 , a condition S1 (RUN) will shift to a condition S2 (SNAP), if the condition that PCICLK is supplied is shown and bus un-operating is detected in this condition. Count actuation of the snap timer 166 is performed in the condition S2 (SNAP). If generating of an interrupt signal and generating of a bus demand signal are detected during count actuation of the snap timer 166, while returning to a condition S1 (RUN), the counted value of the snap timer 166 is returned to initial value. On the other hand, if count actuation of the snap timer 166 is completed in a condition S2 (SNAP), it will shift to a condition S3 (STOP). Supply of PCICLK is suspended in the condition S3 (STOP). If generating of an interrupt signal and generating of a bus demand signal are detected in a condition S3 (STOP), it will return to a condition S1

(RUN).

[0038] The timing of operation when changing in the condition S3 (STOP) from a condition S1 (RUN) by condition S2 (SNAP) course is shown in drawing 7 . That is, if DEASATO [both FRAME# and IRDY#] is detected and it is detected that the bus demand signal and the system-event signal are not generated, the output of 3 input OR gate 164 will serve as "L" level which shows under bus un-operating. Thereby, count actuation of the snap timer 166 is started. PCICLK continues being supplied until the count actuation is completed. And if count actuation is completed, the clock control signal which directs a halt of PCICLK from the clock control signal output circuit 165 will be generated, and supply of PCICLK to each PCI device will be suspended by this.

[0039] Since time amount will be required by resumption of PCICLK supply even if an interrupt signal etc. is generated after that if PCICLK is immediately stopped at the time of bus idle, the fall of system performance will be caused. Therefore, like this example, by carrying out fixed period delay of the halt timing of PCICLK by count actuation of the snap timer 166, the fall of system performance can be prevented now and it becomes possible to perform as many processings as possible collectively within the clock days of supply.

[0040] Moreover, since there is what clock or a required thing after a bus transaction is completed depending on PCI before starting the following event, a

bus demand signal and an interrupt signal can be immediately answered from such a device by using the snap timer 166.

[0041] In addition, instead of stopping PCICLK, although the above explanation explained only the example in the case of stopping PCICLK during bus un-operating, where the frequency is reduced, even if it continues supplying PCICLK, power consumption can be reduced. In this case, the timing of operation which can be set is shown in drawing 8 .

[0042] That is, if DEASATO [both FRAME# and IRDY#] is detected and it is detected that the bus demand signal and the system-event signal are not generated, the output of 3 input OR gate 164 will serve as "L" level which shows under bus un-operating. Thereby, in order to reduce the frequency of PCICLK from the clock control signal output circuit 165, a clock control signal is generated intermittently and the frequency of PCICLK supplied to each PCI device by this is reduced by what [1/]. If a clock demand signal is generated during such slow clock actuation (detection of assertion of FRAME# by the bus supervisory circuit 161, or IRDY#, detection of generating of the bus demand signal by the bus demand detector 162, or detection of generating of the interrupt signal by the system-event detector 163), the output of 3 input OR gate 164 will serve as "H" level which shows under bus actuation. From the clock control signal output circuit 165, the clock control signal which directs the restart

of PCICLK is generated by this, and the frequency of PCICLK supplied to each PCI device by this is returned to the original frequency.

[0043] Moreover, it is used combining such slow clock control and the snap timer 166 of drawing 5, and after under bus un-operating is detected, you may make it drop the frequency of PCICLK after fixed time amount progress.

[0044] Since the usual PCI device is constituted so that it can operate about a part of logic of the interior also in the condition that the bus clock is stopped, and it can generate an interrupt signal and a bus demand signal normally, it is satisfactory, but when a bus clock is stopped depending on a PCI device, there are some which cannot operate at all. Therefore, when such a device is connected, as mentioned above, it is desirable to supply a necessary minimum clock rather than to stop a bus clock completely, and to guarantee actuation of these devices. Thereby, it becomes possible to aim at reduction of power consumption, guaranteeing actuation of a device.

[0045]

[Effect of the Invention] As explained above, according to this invention, by investigating a system state and controlling a halt/restart of a bus clock according to it by detecting the existence of a bus idle condition, a bus demand signal, and an interrupt signal, an unnecessary bus clock can be stopped without mounting special functions, such as the "CLKRUN" protocol, in a PCI device,

and reduction of power consumption can be aimed at. Power save independent of improvement in system performance or the class of PCI device can be realized now by performing use of a snap timer, and control of a bus clock frequency especially.

DESCRIPTION OF DRAWINGS

[Brief Description of the Drawings]

[Drawing 1] The block diagram showing the configuration of the computer system concerning 1 operation gestalt of this invention.

[Drawing 2] Drawing showing an example of the configuration of the bus clock control circuit established in the system of this operation gestalt.

[Drawing 3] The timing chart which shows the clock halt actuation using the bus clock control circuit of drawing 2 .

[Drawing 4] The timing chart which shows the clock restart actuation using the bus clock control circuit of drawing 2 .

[Drawing 5] Drawing showing the 2nd example of a configuration of the bus clock control circuit established in the system of this operation gestalt.

[Drawing 6] Drawing showing the state transition of the bus clock control circuit of drawing 5 .

[Drawing 7] The timing chart which shows the clock halt actuation using the bus clock control circuit of drawing 5 .

[Drawing 8] The timing chart which shows actuation of the bus clock frequency fall processing realized using drawing 2 or the bus clock control circuit of drawing 5 .

[Description of Notations]

2 -- PCI bus

3 -- ISA Bus

11 -- CPU

12 -- Host-PCI bridge

13 -- Memory

14 15 -- PCI peripheral device

16 -- Bus clock control circuit

17 -- Bus clock generation circuit

18 -- PCI-DS bridge

161 -- Bus supervisory circuit

162 -- Bus demand detector

163 -- System-event detector

166 -- Snap timer

【特許請求の範囲】

【請求項 1】 コンピュータシステムのバスに結合される複数の周辺デバイスと、

これら複数の周辺デバイスに前記バス上のトランザクションのタイミングを与えるためのバスクロックを生成するバスクロック生成手段と、

前記バス上のトランザクションを監視し、バスアイドル状態か否かを検出するバスアイドル検出手段と、

前記各周辺デバイスからのバス要求信号および割り込み信号の有無を検出するイベント検出手段と、

前記バスアイドル検出手段および前記イベント検出手段の検出結果に基づいて前記バスクロック生成手段のクロック生成動作を制御するクロック制御手段であって、バスアイドル状態であることが検出され、且つ前記バス要求信号および割り込み信号が発生されてないことが検出されたとき前記バスクロックを停止させるクロック制御手段とを具備することを特徴とするコンピュータシステム。

【請求項 2】 前記クロック制御手段は、前記バスアイドル状態であることが検出され、且つ前記バス要求信号および割り込み信号が発生されてないことが検出されてから、一定時間経過後に前記バスクロックが停止されるように、前記バスクロックを停止させるタイミングを遅延させる手段と、

前記バスアイドル状態の解除、前記バス要求信号、もしくは前記割り込み信号の発生が検出されたとき、前記バスクロックの供給を再開させる手段とを具備することを特徴とする請求項 1 記載のコンピュータシステム。

【請求項 3】 コンピュータシステムのバスに結合される複数のデバイスと、

これら複数のデバイスに前記バス上のトランザクションのタイミングを与えるためのバスクロックを生成するバスクロック生成手段と、

前記バス上のトランザクションを監視し、バスアイドル状態か否かを検出するバスアイドル検出手段と、

前記各デバイスからのバス要求信号および割り込み信号の有無を検出するイベント検出手段と、

前記バスアイドル検出手段および前記イベント検出手段の検出結果に基づいて前記バスクロック生成手段のクロック生成動作を制御するクロック制御手段であって、バスアイドル状態であることが検出され、且つ前記バス要求信号および割り込み信号が発生されてないことが検出されたとき前記バスクロックの周波数を低下させるクロック制御手段とを具備することを特徴とするコンピュータシステム。

【請求項 4】 前記クロック制御手段は、前記バスアイドル状態であることが検出され、且つ前記バス要求信号および割り込み信号が発生されてないことが検出されてから、一定時間経過後に前記バスクロックの周波数が低下されるように、前記バスクロックの周波

数を低下させるタイミングを遅延させる手段と、前記バスアイドル状態の解除、前記バス要求信号もしくは前記割り込み信号の発生が検出されたとき、前記バスクロックの周波数を元の状態に復帰させる手段とを具備することを特徴とする請求項 3 記載のコンピュータシステム。

【発明の詳細な説明】**【0001】**

【発明の属する技術分野】 この発明はコンピュータシステムに関し、特にバスクロック制御機能を有するコンピュータシステムに関する。

【0002】

【従来の技術】 従来、パーソナルコンピュータに使用されるシステムバスとしては、ISA (Industry Standard Architecture) バスや EISA (Extended ISA) バスが主流であったが、最近では、データ転送速度の高速化や、プロセッサに依存しないシステムアーキテクチャの構築のために、PCI (Peripheral Component Interconnect) バスが採用され始めている。

【0003】 PCI バスにおいては、ほとんど全てのデータ転送はブロック転送を基本としており、これら各ブロック転送はバースト転送を用いて実現されている。これにより、PCI バスでは最大 133M バイト/秒 (データバスが 32 ビット幅の時) のデータ転送速度を実現できる。

【0004】 したがって、PCI バスを採用すると、I/O デバイス間、およびシステムメモリと I/O デバイスとの間のデータ転送などを高速に行うことが可能となり、システム性能を高めることができる。

【0005】 このような PCI バスアーキテクチャでは、高速動作の実現のために 33MHz という比較的高速なバスクロックが用いられている。このバスクロックは PCI バス上のトランザクションのタイミングをバス上の各周辺デバイスに提供するためのものであり、システム動作中は常にバス上の周辺デバイスに供給され続けている。

【0006】

【発明が解決しようとする課題】 しかし、このような高速バスクロックは、コンピュータの電力消費量を高める大きな要因の一つとなる。このため、最近では、PCI バスの仕様の中にもバスクロックを停止させるための仕様 ("CLKRUN" プロトコル) が盛り込まれるようになってきている。

【0007】 ところが、この "CLKRUN" プロトコルをシステムに実装するためには、PCI バス上の全てのデバイスが "CLKRUN" プロトコルに対応するための機能を持つことが必要とされる。このため、"CLKRUN" プロトコルに対応してない既存の周辺デバイ

スが存在する場合には“CLKRUN”プロトコルを用いることができず、バスクロックを停止させることはできなかった。

【0008】この発明はこのような点に鑑みてなされたものであり、周辺デバイスに“CLKRUN”プロトコルなどの特別な機能を実装することなくバスクロックを制御できるようにし、容易に消費電力の低減を図ることが可能なコンピュータシステムを提供することを目的とする。

【0009】

【課題を解決するための手段】この発明によるコンピュータシステムは、コンピュータシステムのバスに結合される複数の周辺デバイスと、これら複数の周辺デバイスに前記バス上のトランザクションのタイミングを与えるためのバスクロックを生成するバスクロック生成手段と、前記バス上のトランザクションを監視し、バスアイドル状態か否かを検出するバスアイドル検出手段と、前記各周辺デバイスからのバス要求信号および割り込み信号の有無を検出するイベント検出手段と、前記バスアイドル検出手段および前記イベント検出手段の検出結果に基づいて前記バスクロック生成手段のクロック生成動作を制御するクロック制御手段であって、バスアイドル状態であることが検出され、且つ前記バス要求信号および割り込み信号が発生されていないことが検出されたとき前記バスクロックを停止させるクロック制御手段とを具備することを特徴とする。

【0010】このコンピュータシステムにおいては、バスアイドル状態、バス要求信号および割り込み信号の有無を検出することによりシステム状態が調べられ、バスアイドル状態で、且つバス要求信号および割り込み信号が発生されていないことが検出されたときには、システムアイドルであると判断されてバスクロックが停止される。この場合、たとえばバスアイドル状態であってもバス要求信号または割り込み信号が発生されている限りにおいてはバスクロックは停止されないで、周辺デバイスの動作に影響を与えることはない。よって、周辺デバイスに“CLKRUN”プロトコルなどの特別な機能を実装することなく不要なバスクロックを停止できるようになり、電力消費の低減を図ることができる。

【0011】また、前記クロック制御手段としては、前記バスアイドル状態であることが検出され、且つ前記バス要求信号および割り込み信号が発生されていないことが検出されてから一定時間経過後に前記バスクロックが停止されるように、前記バスクロックを停止させるタイミングを遅延させる手段を含むことが好ましい。

【0012】システムアイドル時にすぐにバスクロックを停止させると、その後に割り込み信号などが発生されてもバスクロックの供給再開までには時間を要するので、システムパフォーマンスの低下を招くことになる。従って、システムパフォーマンスの低下を防止するため

には、クロック供給期間になるべく多くの処理をまとめて実行させるようにすることが必要であり、これはバスクロックを停止させるタイミングを遅延させることによって実現できる。

【0013】また、この発明のコンピュータシステムは、コンピュータシステムのバスに結合される複数のデバイスと、これら複数のデバイスに前記バス上のトランザクションのタイミングを与えるためのバスクロックを生成するバスクロック生成手段と、前記バス上のトランザクションを監視し、バスアイドル状態か否かを検出するバスアイドル検出手段と、前記各デバイスからのバス要求信号および割り込み信号の有無を検出するイベント検出手段と、前記バスアイドル検出手段および前記イベント検出手段の検出結果に基づいて前記バスクロック生成手段のクロック生成動作を制御するクロック制御手段であって、バスアイドル状態であることが検出され、且つ前記バス要求信号および割り込み信号が発生されていないことが検出されたとき前記バスクロックの周波数を低下させるクロック制御手段とを具備することを特徴とする。

【0014】この構成によれば、バスクロックを完全に停止する代わりに、そのバスクロックの周波数が低下される。通常、周辺デバイスは、バスクロックが停止されている状態においてもその内部の一部のロジックについては動作できるように構成されており割り込み信号やバス要求信号を正常に発生できるので問題はないが、周辺デバイスによってはバスクロックが停止されてしまうと全く動作できないものもある。したがって、このようなデバイスが接続されている場合には、バスクロックを完全に停止するのではなく、必要最低限のクロックを供給してそれらデバイスの動作を保証することが必要である。これにより、デバイスの動作を保証しつつ、電力消費の低減を図ることが可能となる。

【0015】

【発明の実施の形態】以下、図面を参照してこの発明の実施形態を説明する。図1には、この発明の一実施形態に係わるコンピュータシステムの構成が示されている。このコンピュータシステムは、バッテリー駆動可能なノートブックタイプまたはラップトップタイプのポータブルコンピュータであり、そのシステムボード上には、プロセッサバス1、PCIバス2、ISAバス3、CPU11、ホスト-PCIブリッジ装置12、主メモリ13、各種PCI周辺デバイス14、15、バスクロック制御回路16、バスクロック生成回路17、PCI-ISAブリッジ装置18、および各種ISA周辺デバイス19、20などが設けられている。

【0016】これらコンポーネントの内、PCIバス2に接続されたPCIデバイスとして機能するもの、すなわちホスト-PCIブリッジ装置12、各種PCI周辺デバイス14、15、PCI-ISAブリッジ装置18

には、バスクロック生成回路17によって生成されたPCIバスクロック(PCI CLK)が供給される。このPCIバスクロック(PCI CLK)は、各PCIデバイスに対してバストランザクションのタイミングを与えるものであり、PCIバス2上のサイクルは全てPCIバスクロック(PCI CLK)に同期して実行される。

【0017】次に、図1の各コンポーネントの機能および構成について説明する。CPU11は、例えば、米インテル社によって製造販売されているマイクロプロセッサ“Pentium”などによって実現されている。このCPU11の入出力ピンに直結されているプロセッサバス1は、64ビット幅のデータバスを有している。主メモリ13は、オペレーティングシステム、デバイスドライバ、実行対象のアプリケーションプログラム、および処理データなどを格納するメモリデバイスであり、複数のDRAMモジュールによって構成されている。

【0018】ホストPCIブリッジ装置12は、プロセッサバス1とPCIバス2との間を繋ぐブリッジLSIであり、PCIバス2のバスマスタの1つとして機能する。このホストPCIブリッジ装置12は、プロセッサバス1とPCIバス2との間で、データおよびアドレスを含むバスサイクルを双方向で変換する機能、およびメモリバスを介して主メモリ13をアクセス制御する機能などを有している。

【0019】PCIバス2はクロック同期型の入出力バスであり、前述したようにPCIバス2上の全てのサイクルはPCIバスクロックに同期して行なわれる。PCIバスクロックの周波数は最大33MHzである。PCIバス2は、時分割的に使用されるアドレス/データバスを有している。このアドレス/データバスは、32ビット幅である。

【0020】PCIバス2上のデータ転送サイクルは、アドレスフェーズとそれに後続する1以上のデータフェーズとから構成される。アドレスフェーズにおいてはアドレスおよび転送タイプが出力され、データフェーズでは8ビット、16ビット、24ビットまたは32ビットのデータが出力される。

【0021】PCIデバイス14、15は例えばグラフィクスコントローラ、PCカード(カードバス)コントローラ、IrDAコントローラ、SCSIコントローラなどであり、ホストPCIブリッジ装置12と同様にPCIバス2のバスマスタとして機能する。

【0022】PCI-ISAブリッジ装置16は、PCIバス2とISAバス3との間を繋ぐブリッジLSIであり、PCIデバイスの1つとして機能する。ISAバス3上のISAデバイス19、20は、例えば、HDD、システムタイマ、キーボードコントローラなどである。

【0023】バスクロック制御回路16は各PCIデバイスに対するPCIバスクロック(PCI CLK)の供

給の停止/再開を制御するためのものであり、そのPCIバスクロック(PCI CLK)の停止/再開の制御は、バスアイドル信号、バス要求信号、およびシステムイベント信号に基づいて行われる。バスアイドル信号はPCIバス2の動作状態、つまりバス動作中(バスサイクル実行中)であるか否かを示す。バス要求信号は、バスマスタがPCIバス2の使用権をPCIバスアービタに要求するための信号である。システムイベント信号は、割り込み信号などシステム内で何らかのイベントが発生したことを示すものである。

【0024】バスクロック制御回路16は、バスアイドル信号によるPCIバスのアイドルを検出し、バス要求信号、およびシステムイベント信号がなければ、クロック制御信号によりバスクロック生成回路17を制御してバスクロックを停止させる。また、バスクロック停止中に、バス要求信号またはシステムイベント信号が発生した場合には、バスクロック制御回路16は、クロック制御信号によりバスクロック生成回路17を制御してバスクロックの供給を再開させる。

【0025】図2には、バスクロック制御回路16の具体的な構成例が示されている。図示のように、バスクロック制御回路16は、バス監視回路161、バス要求検出回路162、システムイベント検出回路163、ORゲート164、およびクロック制御信号出力回路165から構成されている。

【0026】バス監視回路161は前述のバスアイドル信号を生成するためのものであり、PCIバス2上に定義されたフレーム信号FRAME#およびイニシエータレディー信号IRDY#を用いてPCIバス2上のランザクションを監視し、バス非動作であることを検出したときにバスアイドル信号を“H”レベルにアサートし、バス動作中であることを検出したときにはバスアイドル信号を“L”レベルにデアサートする。

【0027】ここで、FRAME#は、ランザクションの開始とその期間を示すために現在のマスタによってドライブされる信号である。FRAME#がデアサートされた時、ランザクションが最後のデータフェーズであることを示す。IRDY#は現在のマスタによってドライブされる信号であり、ライトサイクルにおいては、マスタがバス上に確定データを出力したことを示すためにアサートされ、リードサイクルにおいては、マスタがデータを受信する準備ができたことを示すためにアサートされる。

【0028】これらFRAME#およびIRDY#が共にデアサートされているとき、バス監視回路161はバスアイドル状態であることを検出し、バスアイドル信号を“H”レベルにアサートする。

【0029】バス要求検出回路162は、PCIデバイスそれぞれからPCIバスアービタに入力される全てのバスリクエスト信号REQ1#~REQn#を監視し、

REQ1#～REQn#のいずれかがアサートされているときバス要求信号を“H”レベルにアサートする。

【0030】システムイベント検出回路は、各PCIデバイスから割り込みコントローラに入力される割り込み信号INTA-D、各ISAデバイスから割り込みコントローラに入力される割り込み信号IRQ0-15、さらにマスク不能割り込み信号NMI、システム管理割り込み信号SMIについての監視を行い、いずれかの割り込み信号が発生されているときシステムイベント信号を“H”レベルにアサートする。

【0031】バス監視回路161からのバスアイドル信号は、3入力ORゲート164の第1入力に反転入力される。また、バス要求検出回路162およびシステムイベント検出回路163からのバス要求信号およびシステムイベント信号は、そのまま3入力ORゲート164の第2入力および第3入力にそれぞれ入力される。3入力ORゲート164の出力はバス動作中(“H”レベル)／バス非動作中(“L”レベル)を示すものであり、クロック制御信号出力回路165に入力される。

【0032】クロック制御信号出力回路165は、3入力ORゲート164の出力に基づいてPCICLKの停止／再開を指示するクロック制御信号を発生する。次に、図3および図4を参照して、本実施形態の動作を説明する。

【0033】図3のタイミングチャートはバスクロック(PCICLK)を停止させる場合のタイミングである。すなわち、FRAME#およびIRDY#が共にデアサートされたことが検出され、且つバス要求信号およびシステムイベント信号が発生されていないことが検出されると、3入力ORゲート164の出力はバス非動作中を示す“L”レベルとなる。これにより、クロック制御信号出力回路165からはPCICLKの停止を指示するクロック制御信号が発生され、これによって各PCIデバイスに対するPCICLKの供給が停止される。

【0034】このPCICLKの供給停止期間中に、クロック要求信号が発生されると(バス監視回路161によるFRAME#またはIRDY#のアサートの検出、バス要求検出回路162によるバス要求信号の発生の検出、またはシステムイベント検出回路163による割り込み信号の発生の検出)、3入力ORゲート164の出力はバス動作中を示す“H”レベルとなる。これにより、クロック制御信号出力回路165からはPCICLKの再開を指示するクロック制御信号が発生され、これによって各PCIデバイスに対するPCICLKの供給が再開される。

【0035】このように、図2の構成によれば、バスアイドル状態、バス要求信号および割り込み信号の有無を検出することによりシステム状態が調べられ、バスアイドル状態で、且つバス要求信号および割り込み信号が発生されていないことが検出されたときには、システムアイ

ドルであると判断されてPCICLKが停止される。この場合、たとえばバスアイドル状態であってもバス要求信号または割り込み信号が発生されている限りにおいてはPCICLKは停止されないで、各PCIデバイスの動作に影響を与えることはない。また、PCICLKの停止中にPCIデバイスからバス要求信号や割り込み信号が発生されると、PCICLKの供給が再開され、PCICLKに同期した正常なバストランザクションを行うことが可能となる。よって、PCIデバイスに“CLKRUN”プロトコルなどの特別な機能を実装することなく不要なバスクロックを停止できるようになり、電力消費の低減を図ることができる。

【0036】図5には、バスクロック制御回路16の第2の構成例が示されている。ここでは、図2の構成に加え、スナップタイマ166が設けられている。このスナップタイマ166は、PCICLKの停止タイミングを一定期間遅らせるためのものであり、3入力ORゲート164の出力がバス非動作中を示す“L”レベルとなつてからカウント動作を開始し、所定のカウンタ値に達したときにそれをクロック制御信号出力回路165に通知する。これにより、PCICLKの停止を指示するクロック制御信号は、スナップタイマ166のカウント時間だけ遅れて発生されることになる。スナップタイマ166のカウント値は、たとえばバスクロック制御回路16内にカウンタ値を設定するためのコンフィグレーションレジスタを設け、そこに希望する遅れ時間に相当するカウンタ値をソフトウェア的に設定することによって、プログラマブルにすることができる。

【0037】図6には、図5のバスクロック制御回路16の状態遷移の様子が示されている。図6において、状態S1(RUN)はPCICLKが供給されている状態を示し、この状態でバス非動作が検知されると、状態S2(SNAP)に移行される。状態S2(SNAP)では、スナップタイマ166のカウント動作が行われる。スナップタイマ166のカウント動作中に割り込み信号の発生やバス要求信号の発生が検出されると、状態S1(RUN)に復帰されると共に、スナップタイマ166のカウント値は初期値に戻される。一方、状態S2(SNAP)においてスナップタイマ166のカウント動作が完了すると、状態S3(STOP)に移行される。状態S3(STOP)では、PCICLKの供給が停止される。状態S3(STOP)において割り込み信号の発生やバス要求信号の発生が検出されると、状態S1(RUN)に復帰される。

【0038】図7には、状態S2(SNAP)経由で状態S1(RUN)から状態S3(STOP)に遷移するときの動作タイミングが示されている。すなわち、FRAME#およびIRDY#が共にデアサートされたことが検出され、且つバス要求信号およびシステムイベント信号が発生されていないことが検出されると、3入力OR

ゲート 1 6 4 の出力はバス非動作中を示す“L”レベルとなる。これにより、スナップタイマ 1 6 6 のカウント動作が開始される。そのカウント動作が完了するまでは P C I C L K は供給され続ける。そして、カウント動作が完了すると、クロック制御信号出力回路 1 6 5 から P C I C L K の停止を指示するクロック制御信号が発生され、これによって各 P C I デバイスに対する P C I C L K の供給が停止される。

【0 0 3 9】バスアイドル時にすぐに P C I C L K を停止させると、その後に割り込み信号などが発生されても P C I C L K の供給再開までには時間を要するので、システムパフォーマンスの低下を招くことになる。従って、本例のように、P C I C L K の停止タイミングをスナップタイマ 1 6 6 のカウント動作によって一定期間遅延させることにより、システムパフォーマンスの低下を防止できるようになり、クロック供給期間内になるべく多くの処理をまとめて実行させることが可能となる。

【0 0 4 0】また、P C I によってはバストランザクションが完了してから、つぎのイベントを起こすまでに何クロックか必要なものがあるので、スナップタイマ 1 6 6 を使用することによって、そのようなデバイスからのバス要求信号や割り込み信号に即座に応答できるようになる。

【0 0 4 1】なお、以上の説明では、バス非動作中に P C I C L K を停止する場合の例についてのみ説明したが、P C I C L K を停止する代わりに、その周波数を低下させた状態で P C I C L K を供給し続けるようにしても電力消費を低減することができる。この場合における動作タイミングを図 8 に示す。

【0 0 4 2】すなわち、F R A M E # および I R D Y # が共にデアサートされたことが検出され、且つバス要求信号およびシステムイベント信号が発生されていないことが検出されると、3 入力 O R ゲート 1 6 4 の出力はバス非動作中を示す“L”レベルとなる。これにより、クロック制御信号出力回路 1 6 5 からは P C I C L K の周波数を低下させるためにクロック制御信号が断続的に発生され、これによって各 P C I デバイスに供給される P C I C L K の周波数は何分の 1 かに低減される。このようなスロークロック動作中に、クロック要求信号が発生されると（バス監視回路 1 6 1 による F R A M E # または I R D Y # のアサートの検出、バス要求検出回路 1 6 2 によるバス要求信号の発生の検出、またはシステムイベント検出回路 1 6 3 による割り込み信号の発生の検出）、3 入力 O R ゲート 1 6 4 の出力はバス動作中を示す“H”レベルとなる。これにより、クロック制御信号出力回路 1 6 5 からは P C I C L K の再開を指示するクロック制御信号が発生され、これによって各 P C I デバイスに供給される P C I C L K の周波数は元の周波数に戻される。

【0 0 4 3】また、このようなスロークロック制御と図

5 のスナップタイマ 1 6 6 とを組み合わせ使用し、バス非動作中が検出されてから一定時間経過後に P C I C L K の周波数を落とすようにしてもよい。

【0 0 4 4】通常の P C I デバイスは、バスクロックが停止されている状態においてもその内部の一部のロジックについては動作できるように構成されており割り込み信号やバス要求信号を正常に発生できるので問題はないが、P C I デバイスによってはバスクロックが停止してしまうと全く動作できないものもある。したがって、このようなデバイスが接続されている場合には、前述したように、バスクロックを完全に停止するのではなく、必要最低限のクロックを供給してそれらデバイスの動作を保証することが好ましい。これにより、デバイスの動作を保証しつつ、電力消費の低減を図ることが可能となる。

【0 0 4 5】

【発明の効果】以上説明したように、この発明によれば、バスアイドル状態、バス要求信号および割り込み信号の有無を検出することによりシステム状態を調べ、それに応じてバスクロックの停止／再開を制御することにより、P C I デバイスに“C L K R U N”プロトコルなどの特別な機能を実装することなく不要なバスクロックを停止できるようになり、電力消費の低減を図ることができる。特に、スナップタイマの使用およびバスクロック周波数の制御を行うことにより、システムパフォーマンスの向上や P C I デバイスの種類に依存しないパワーセーブを実現できるようになる。

【図面の簡単な説明】

【図 1】この発明の一実施形態に係るコンピュータシステムの構成を示すブロック図。

【図 2】同実施形態のシステムに設けられたバスクロック制御回路の構成の一例を示す図。

【図 3】図 2 のバスクロック制御回路を用いたクロック停止動作を示すタイミングチャート。

【図 4】図 2 のバスクロック制御回路を用いたクロック再開動作を示すタイミングチャート。

【図 5】同実施形態のシステムに設けられたバスクロック制御回路の第 2 の構成例を示す図。

【図 6】図 5 のバスクロック制御回路の状態遷移を示す図。

【図 7】図 5 のバスクロック制御回路を用いたクロック停止動作を示すタイミングチャート。

【図 8】図 2 または図 5 のバスクロック制御回路を用いて実現されるバスクロック周波数低下処理の動作を示すタイミングチャート。

【符号の説明】

2 … P C I バス

3 … I S A バス

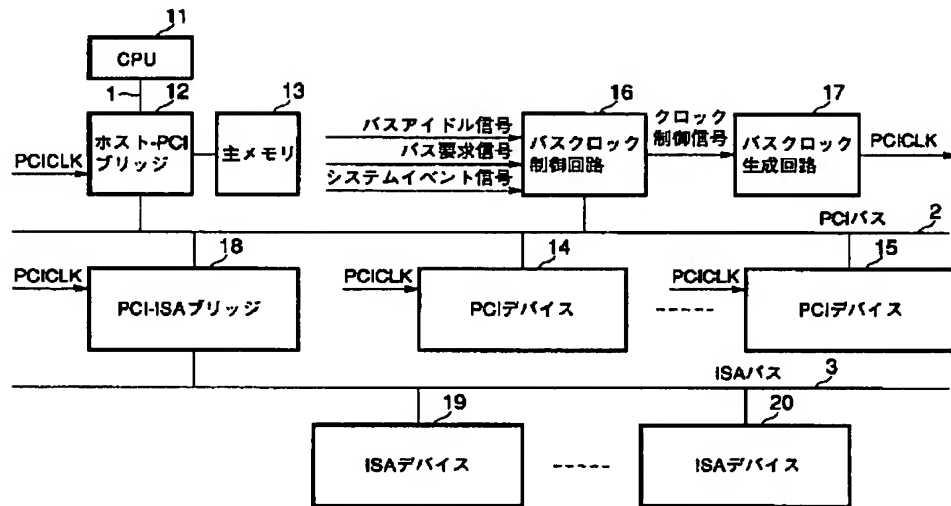
1 1 … C P U

1 2 … ホスト - P C I ブリッジ

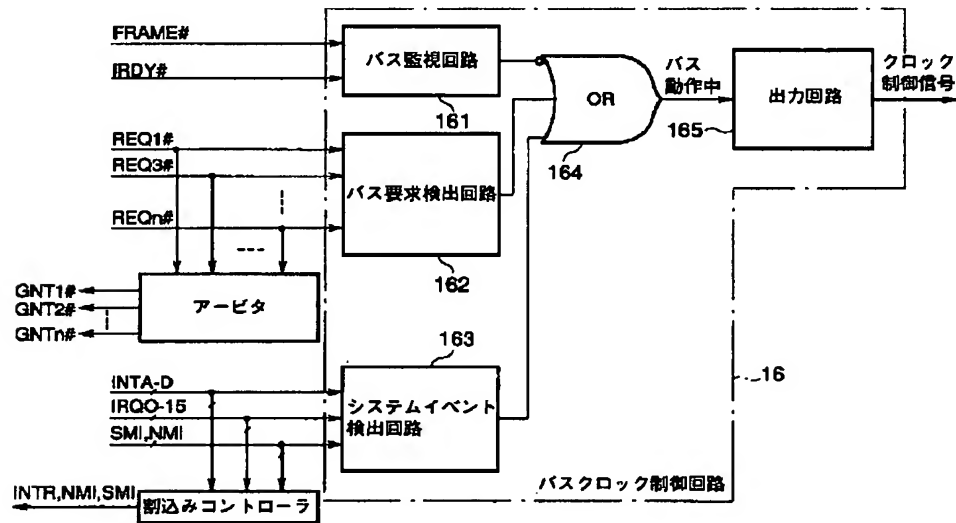
13…メモリ
 14, 15…PCI周辺デバイス
 16…バスクロック制御回路
 17…バスクロック生成回路
 18…PCI-D Sブリッジ

161…バス監視回路
 162…バス要求検出回路
 163…システムイベント検出回路
 166…スナップタイマ

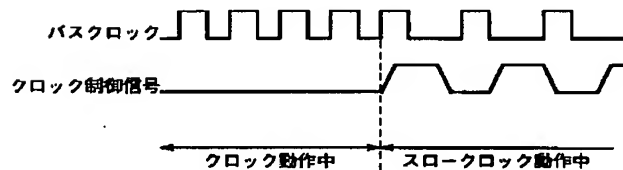
【図1】



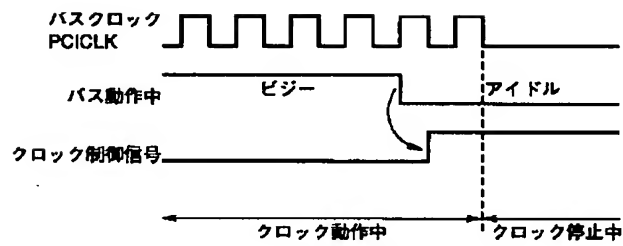
【図2】



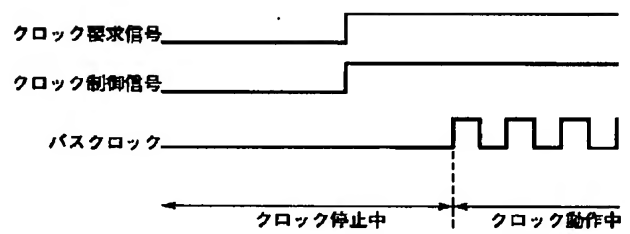
【図8】



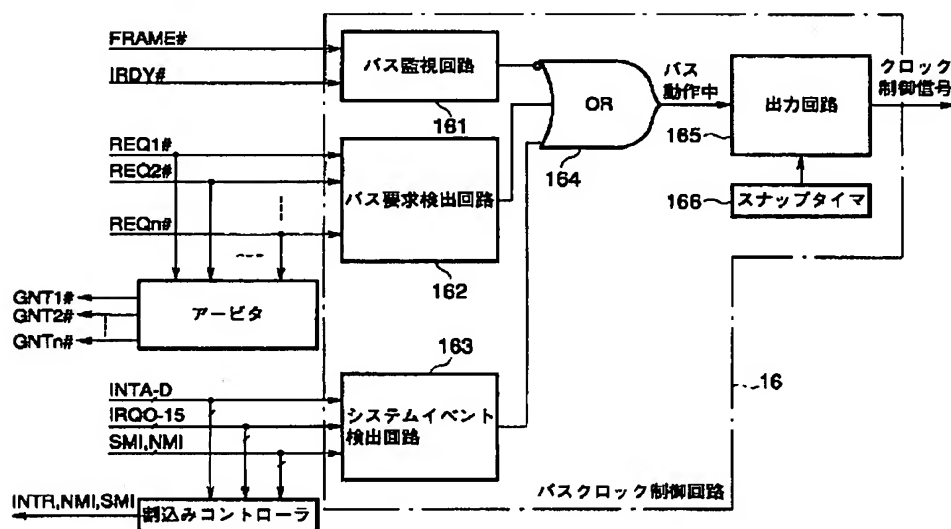
【図 3】



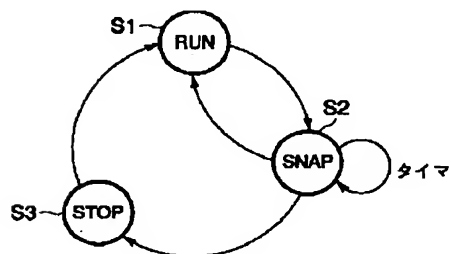
【図 4】



【図 5】



【図 6】



【図 7】

